Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Институт Информационных Технологий и Анализ Данных

Кафедра вычислительной техники

**Название работы** – Разработка консольного приложения

Отчет по лабораторной работе № 1

Вариант 2

по дисциплине Объектно-ориентированное программирование

Выполнил

Студент, номер группы ИСМб-19-1 \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Д. Д. Солопов

Дата: \_\_\_\_\_\_\_\_\_\_

Принял

Должность \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Т.В. Маланова

Дата: \_\_\_\_\_\_\_\_\_\_

Иркутск 2020

Содержание

[1 Постановка задачи 3](#_Toc54960103)

[2 Таблица спецификаций 3](#_Toc54960104)

[3 Блок-схемы алгоритмов решения задач 5](#_Toc54960105)

[4 Таблица тестов 10](#_Toc54960106)

[5 Листинг исходного кода 17](#_Toc54960107)

[6 Заключение 20](#_Toc54960108)

[7 Список использованных источников 21](#_Toc54960109)

1 Постановка задачи

**a.** Определите, является ли одна строка подстрокой другой.

**b.** Найти длину указанного слова в предложении.

**c.** Выведите из строки, содержащей слова, разделенные пробелами и запятыми, центральное слово (если в предложении два централь-ных слова, выведите оба).

2 Таблица спецификаций

Таблица 1 – Таблица спецификаций для задания **a**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Имя** | **Назначение** | **Тип** | **ОДЗ** |
| **int main() – точка входа в программу** | | | | |
| 1 | text | Входные данные - исходная строка | String | - |
| 2 | substring | Входные данные – искомая подстрока | String | - |
| 3 | answer | Выходные данные – строка содержащий ответ на вопрос “присутствует ли подстрока substring в строке text” | String | - |

Таблица 2 – Таблица спецификаций для задания **b**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Имя** | **Назначение** | **Тип** | **ОДЗ** |
| **int main() – точка входа в программу** | | | | |
| 1 | text | Входные данные - исходная строка | String | - |
| 2 | number | Входные данные – номер слова в строке text, длину которого необходимо найти | int | (0; размер строки text) |
| 3 | answer | Выходные данные – строка содержащая ответ на вопрос “какую длину имеет слово имеющее в предложении порядковый номер number?” | String | - |

Таблица 3 – Таблица спецификаций для задания **c**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Имя** | **Назначение** | **Тип** | **ОДЗ** |
| **int main() – точка входа в программу** | | | | |
| 1 | text | Входные данные - исходная строка | String | - |
| 2 | answer | Выходные данные – строка содержащая центральное(-ые) предложение(-ия) в строке text | String | - |

3 Блок-схемы алгоритмов решения задач

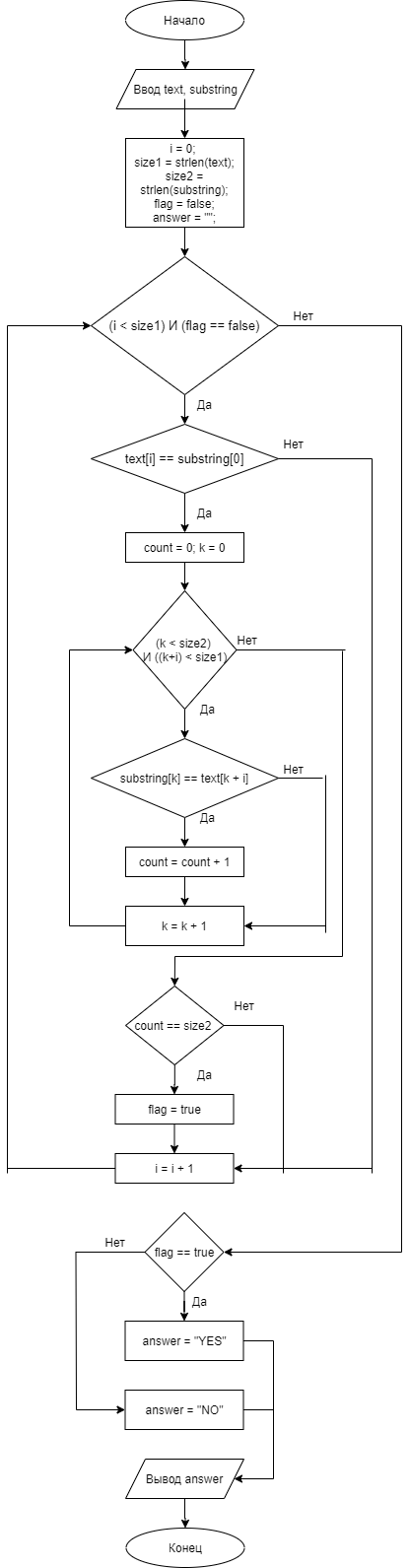


Рисунок 1 – Блок-схема алгоритма решения задания a

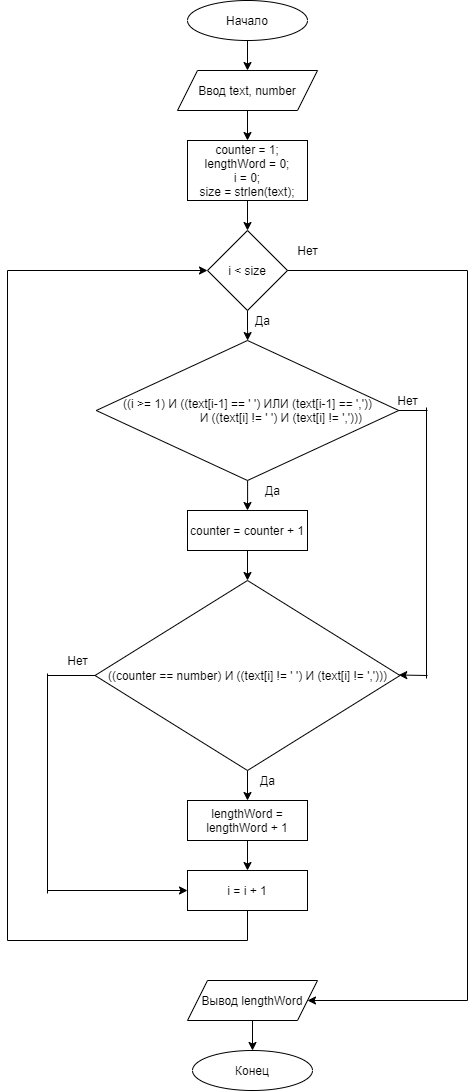


Рисунок 2 – Блок-схема алгоритма решения заданияb

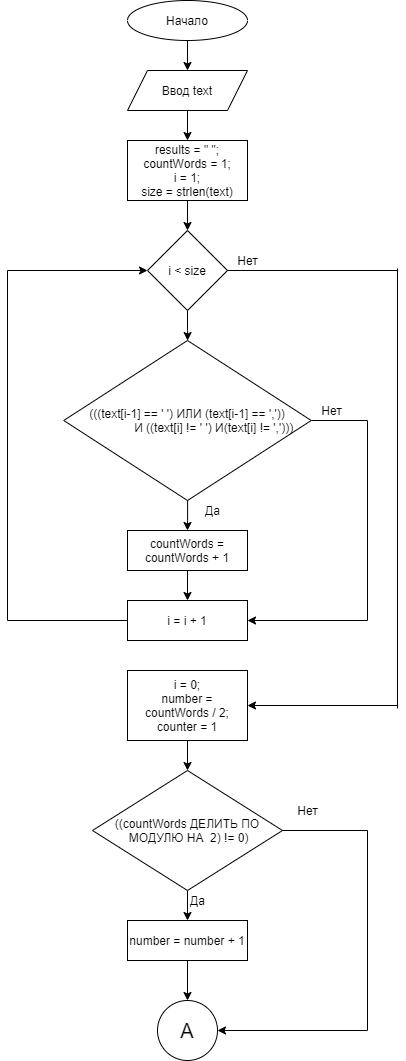


Рисунок 3 – Первая часть блок схемы алгоритма решения задания c

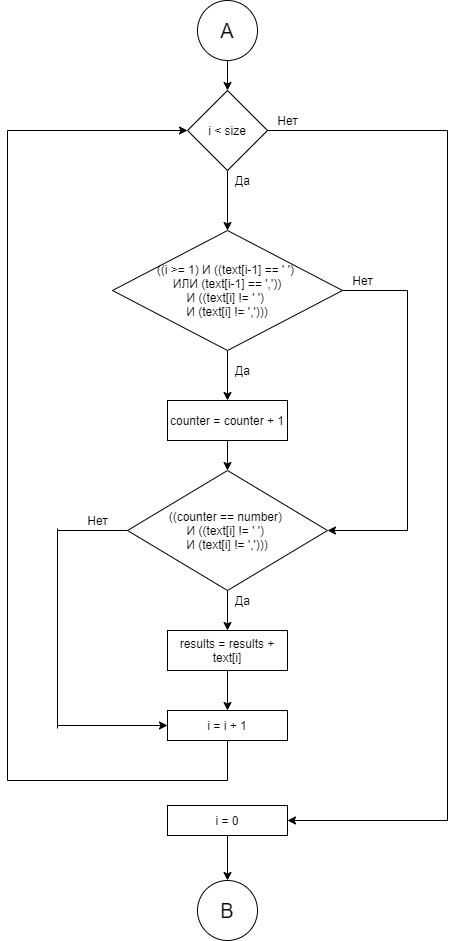


Рисунок 4 – Вторая часть блок схемы алгоритма решения задания c

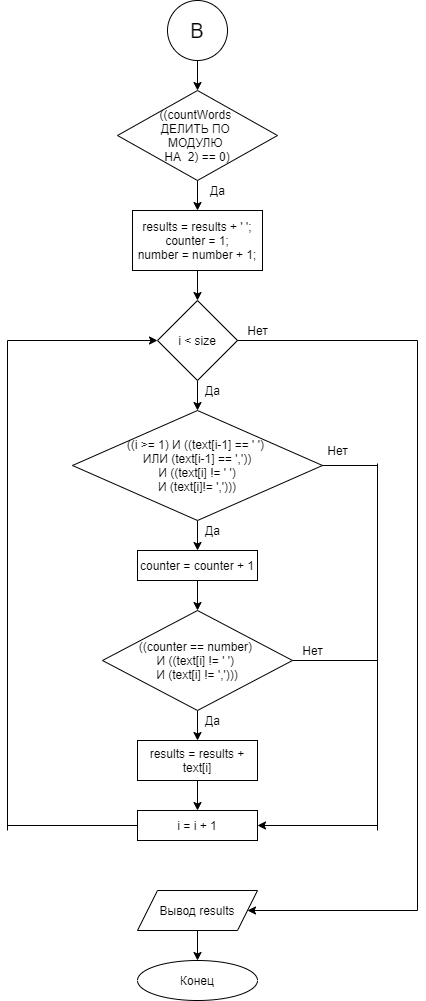


Рисунок 5 – Третья часть блок схемы алгоритма решения задания c

4 Таблица тестов

Таблица 4 – Таблица тестов для решения задачи a

|  |  |  |  |
| --- | --- | --- | --- |
| **N теста** | **Назначение теста** | **Входные данные для теста** | **Выходные данные для теста** |
| 1 | Ввод осмысленного текста(подстрока найдена) | text = “Здравствуйте уважаемые коллеги!”  substring = “коллеги” | Подстрока "коллеги" присутствует в строке "Здравствуйте уважаемые коллеги!" (Рис.6) |
| 2 | Ввод не осмысленного текста(подстрока найдена) | text = “helloa dfwr tdght n tonight”  substring = “ht” | Подстрока "ht" присутствует в строке "helloa dfwr tdght n tonight" (Рис.7) |
| 3 | Ввод текста содержащего различные символы(подстрока не найдена) | text = “с№;ырфкз c#sharp124 hello 123 %!!!sharp”  substring = “%!!!sharp” | Подстрока "%!!!sharp" присутствует в строке "с№;ырфкз c#sharp124 hello 123 %!!!sharp"  (Рис. 8) |
| 4 | Ввод осмысленного текста(подстрока не найдена) | text = “Здравствуйте! Товарищи и коллеги!”  substring = “автомобиль” | Подстрока "автомобиль" не присутствует в строке "Здравствуйте! Товарищи и коллеги!"  (Рис.9) |

Продолжение таблицы 4

|  |  |  |  |
| --- | --- | --- | --- |
| 5 | Ввод не осмысленного текста(подстрока не найдена) | text = “uqhuiqh oiwehu hwuhiuhw uop”  substring = “uops” | Подстрока "uops" не присутствует в строке "uqhuiqh oiwehu hwuhiuhw uop" (Рис.10) |
| 6 | Ввод текста содержащего различные символы(подстрока не найдена) | text = “afh1982y419(\*!&( (\* Y( iG 8G8 &^^! g\*G 8^T ^g”  substring = “!@#$%^” | Подстрока "!@#$%^" не присутствует в строке "afh1982y419(\*!&( (\* Y( iG 8G8 &^^! g\*G 8^T ^g" (Рис.11) |

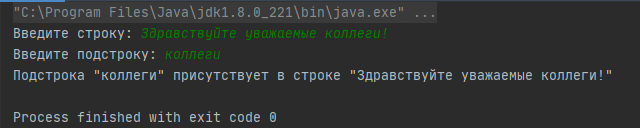


Рисунок 6 – Результат теста №1 из таблицы тестов №4

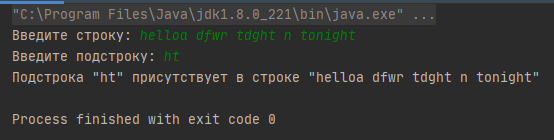


Рисунок 7 – Результат теста №2 из таблицы тестов №4

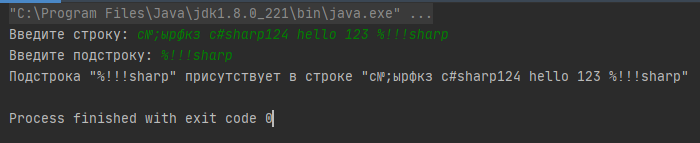


Рисунок 8 – Результат теста №3 из таблицы тестов №4

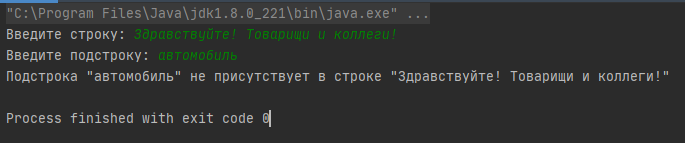


Рисунок 9 – Результат теста №4 из таблицы тестов №4

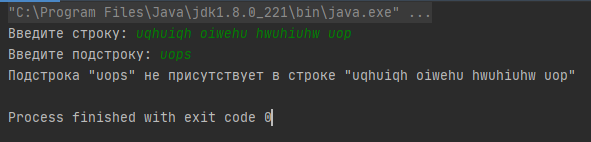


Рисунок 10 – Результат теста №5 из таблицы тестов №4

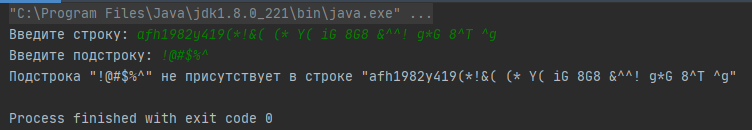


Рисунок 11 – Результат теста №6 из таблицы тестов №4

Таблица 5 – Таблица тестов для решения задачи b

|  |  |  |  |
| --- | --- | --- | --- |
| **N теста** | **Назначение теста** | **Входные данные для теста** | **Выходные данные для теста** |
| 1 | Ввод осмысленного текста(слово с введёным порядковым номером существует) | text = “ Hello, my world! Hello, my car!”  number = 1 | Слово с порядковым номером 4 имеет длину 5 (Рис.12) |
| 2 | Ввод осмысленного текста(слово с введёным порядковым номером не существует) | text = “ Hello, my world! Hello, my car!”  number = 15 | Слово с порядковым номером 15 имеет длину 0 (Рис.13) |
| 3 | Ввод текста содержащего различные символы(слово с введённым порядковым номером существует) | text = “с№;ырфкз c#sharp124 hello 123 %!!!sharp”  number = 5 | Слово с порядковым номером 5 имеет длину 9 (Рис. 14) |
| 4 | Ввод текста содержащего различные символы(слово с введённым порядковым номером не существует) | text = “ hwurha uawuhr iuahw,aw oaiwi aw,t a,wtm al;wkt  number = 20 | Слово с порядковым номером 20 имеет длину 0 (Рис.15) |

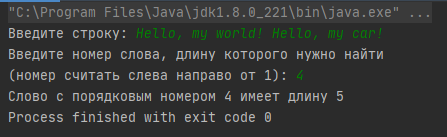


Рисунок 12 - Результат теста №1 из таблицы тестов №5

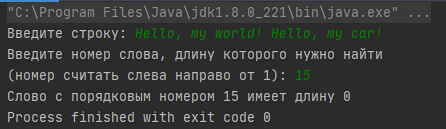


Рисунок 13 - Результат теста №2 из таблицы тестов №5

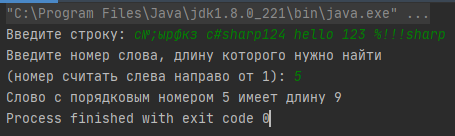


Рисунок 14 – Результат теста №3 из таблицы тестов №5

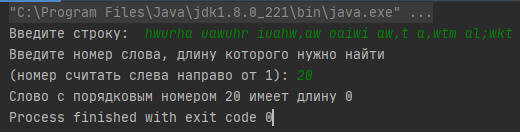


Рисунок 15 - Результат теста №4 из таблицы тестов №5

Таблица 6 – Таблица тестов для решения задачи c

|  |  |  |  |
| --- | --- | --- | --- |
| **N теста** | **Назначение теста** | **Входные данные для теста** | **Выходные данные для теста** |
| 1 | Ввод текста с корректными разделителями | text = “It's my computer!” | Центральное слово/слова в предложении: my (Рис.16) |
| 2 | Ввод текста с пробелами и запятыми | text = “Hello, my, world!, Hello, my, car!” | Центральное слово/слова в предложении: world! Hello (Рис.17) |
| 3 | Ввод текста с некорректными разделителями | text = “ It's , , , , , , , ,, , , my , ,, , , , , , , , , computer! “ | Центральное слово/слова в предложении: my  (Рис.18) |
| 4 | Ввод текста с пробелами | text = “abcd,efgh,phgf,wrtf,ghtr” | Центральное слово/слова в предложении: phgf  (Рис.19) |

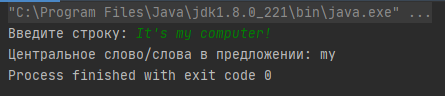


Рисунок 16 - Результат теста №1 из таблицы тестов №6

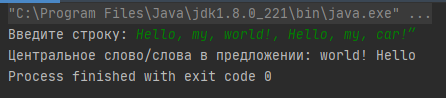


Рисунок 17 - Результат теста №2 из таблицы тестов №6

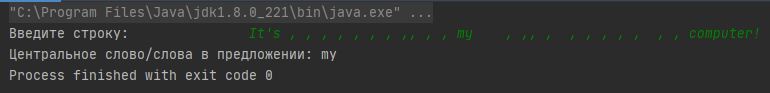


Рисунок 18 - Результат теста №3 из таблицы тестов №6

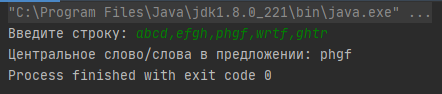


Рисунок 19 - Результат теста №4 из таблицы тестов №6

5 Листинг исходного кода

Листинг исходного кода, для задания a:

package com.company;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

/\*Вариант 2

a. Определите, является ли одна строка подстрокой другой.

\*/

System.out.print("Введите строку: ");

Scanner scan = new Scanner(System.in);

String text = scan.nextLine();

System.out.print("Введите подстроку: ");

String substring = scan.nextLine();

text = text.trim(); //удаление пробелом из строки

substring = substring.trim();

boolean flag = false;

String answer = "";

for(int i = 0; (i < text.length()) && (!flag); i++)

if(text.charAt(i) == substring.charAt(0)){

int count = 0;

for(int k = 0; k < substring.length(); k++)

if(substring.charAt(k) == text.charAt(k + i))

count++;

if(count == substring.length())

flag = true;

}

if(flag)

answer = "Подстрока " + substring + " присутствует в строке " + text;

else

answer = "Подстрока " + substring + " не присутствует в строке " + text;

System.out.println(answer);

scan.close();

return;

}

}

Листинг исходного кода для задания b:

package com.company;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

/\*Вариант 2

b. Найти длину указанного слова в предложении.\*/

System.out.print("Введите строку: ");

Scanner scan = new Scanner(System.in);

String text = scan.nextLine();

System.out.print("Введите номер слова, длину которого нужно найти\n(номер считать слева направо от 1): ");

int number = scan.nextInt();

while((number <= 0) || (number >= text.length())){

System.out.print("Введите корректный номер слова: ");

number = scan.nextInt();

}

text = text.trim();

int counter = 1;

int lengthWord = 0;

for(int i = 0; i < text.length(); i++){

if((i >= 1) && ((text.charAt(i-1) == ' ') || (text.charAt(i-1) == ','))

&& ((text.charAt(i) != ' ') && (text.charAt(i) != ',')))

counter++;

if((counter == number) && ((text.charAt(i) != ' ') && (text.charAt(i) != ',')))

lengthWord++;

}

String answer = "Слово с порядковым номером " + String.valueOf(number) + " имеет длину " + String.valueOf(lengthWord);

System.out.print(answer);

scan.close();

return;

}

}

Листинг исходного кода для задания c:

package com.company;

import java.util.Scanner;

public class Main {public static void main(String[] args) {

//Вариант 2

//c. Выведите из строки, содержащей слова, разделенные пробелами и запятыми, центральное слово

// (если в предложении два централь-ных слова, выведите оба).

Scanner scan = new Scanner(System.in);

System.out.print("Введите строку: ");

String text = scan.nextLine();

String results = "";

int countWords = 1;

text = text.trim();

for(int i = 1; i < text.length(); i++)

if(((text.charAt(i - 1) == ' ') || (text.charAt(i-1) == ','))

&& ((text.charAt(i) != ' ') && (text.charAt(i) != ',')))

countWords++;

int number = (int)(countWords / 2), counter = 1;

if((countWords % 2) != 0)

number++;

for(int i = 0; i < text.length(); i++){

if((i >= 1) && ((text.charAt(i-1) == ' ') || (text.charAt(i-1) == ','))

&& ((text.charAt(i) != ' ') && (text.charAt(i) != ',')))

counter++;

if((counter == number) && ((text.charAt(i) != ' ') && (text.charAt(i) != ',')))

results += text.charAt(i);

}

if((countWords % 2) == 0){

results += ' ';

counter = 1;

number++;

for(int i = 0; i < text.length(); i++){

if((i >= 1) && ((text.charAt(i-1) == ' ') || (text.charAt(i-1) == ','))

&& ((text.charAt(i) != ' ') && (text.charAt(i) != ',')))

counter++;

if((counter == number) && ((text.charAt(i) != ' ') && (text.charAt(i) != ',')))

results += text.charAt(i);

}

}

scan.close();

String answer = "Центральное слово/слова в предложении: " + results;

System.out.print(answer);

return;

}

}

6 Заключение

Знакомство с платформой Java, а также освоение на практике основных конструкций языка программирования Java, прошло успешно. Были написаны алгоритмы реализующие решение задач согласно индивидуальному варианту, алгоритмам сопоставлены блок-схемы, а также составлены тесты, для проверки работоспособности кода и корректности выходного результата. Все тесты были успешно пройдены, ошибок или не корректной обработки данных не выявлено.

7 Список использованных источников

1. Java. Экспресс-курс [электронный ресурс] // Сайт Александра Климова [сайт], URL: <http://developer.alexanderklimov.ru/android/java/java.php> (дата обращение 12.09.20)

2. API Specification for the Java 7 SE. [официальный сайт] URL: <http://docs.oracle.com/javase/7/docs/api/> (дата обращения 12.09.20)

3. Гради Буч, Роберт А. Максимчук, Майкл У. Энгл, Бобби Дж. Янг, Джим Коналлен, Келли А. Хьюстон. Объектно-ориентированны анализ и проек-тирование с примерами приложений. Третье издание. М.: "Вильямс", 2010, -720 с.

4. Хабибуллин И.Ш. Java 7: для программистов / И. Ш. Хабибуллин. – Санкт-Петербург : БХВ–Петербург, 2014. – 768 с.

5. Васильев А. Н. Java. Объектно-ориентированное программирование: для магистров и бакалавров. Базовый курс по объектно-ориентированному программированию / А. Н. Васильев . – СПб.: Питер, 2012. – 395 с.